

# Практикум 3 курс. Задание 3

## Постановка задачи

Нам необходимо по данной в CSV файлах информации о:

1. закупках (поставки яблок и карандашей два раза в месяц)
2. продажах (лог транзакций, по записи на каждую проданную позицию)
3. инвентаре (месячные данные общего количества яблок и карандашей на складе)

проанализировать и вывести в CSV файлы:

1. Данные о состоянии склада в конце каждого дня после того как все поставки и продажи были совершены. Состояние склада строится на основе месячных данных об инвентаре. В этой статистике нет никакой возможности узнать объем сворованного товара.
2. Месячные данные о количестве сворованного товара
3. Агрегированные данные об объемах продаж и количестве сворованной продукции по штату и году

Набор входных и выходных данных для тестирования вашего решения доступен по следующей ссылке: <https://console.cloud.google.com/storage/browser/artempyanykh-cmc-prac-task3-seed17>

Для загрузки входных и выходных данных для тестирования на компьютер необходимо установить Google Cloud SDK. Скачиваем их с помощью команды:

```
gsutil -m cp -r gs://artem-pyanykh-cmc-prac-task3-seed17/out ./
```

В ней мы:

1. Командой **cp** создаем локальную копию папки **out** на нашем компьютере;
2. Опцией **-m** проводим скачивание параллельно из-за большого количества файлов;
3. Опция **-r** позволяет сделать это рекурсивно, чтобы вместе с папкой загрузилось все ее содержимое.

В скачанной папке содержатся 3 типа файлов для каждого магазина в штате:

1. (имя штата)-(название магазина)-supply.csv - информация по поставкам (1го и 15го числа месяца)
2. (имя штата)-(название магазина)-sell.csv - транзакции на каждый проданный продукт
3. (имя штата)-(название магазина)-inventory.csv - месячные данные общего количества яблок и карандашей на складе.

Выходные данные по первому пункту задания мы сохраняем в файле `./result/(имя штата)-(название магазина)-daily.csv`. Например, `./result/MS-b1-daily.csv`

Выходные данные по второму пункту задания в файле `./result/(имя штата)-(название магазина)-steal.csv`. Например, `./result/MS-b1-steal.csv`

По третьему - ./result/allstat.csv

## Программный подход к решению задачи

Для выполнения задания необходимо подключить библиотеки csv и datetime. Первая из них необходима для работы с файлами формата .csv, с помощью функций этой библиотеки мы считываем и записываем данные. Из второй библиотеки мы используем два класса – date и timedelta. Date используется для удобства работы с датами, а timedelta необходим для того, чтобы мы могли изменять объект класса date на один день.

Для хранения данных по третьему пункту задания используется массив resAllStat.

В программе используется несколько вложенных циклов, первый перемещает по штатам, второй - по магазинам штата, третий – по месяцам, четвертый - по дням месяца, пятый - по строкам файла 'sell.csv' до завершения дня, то есть по проданным товарам в течении дня.

В самом внутреннем цикле мы считываем информацию о проданном товаре, анализируем ее и вычитаем из имеющегося на складе соответствующего товара единицу. В результате работы внутреннего цикла мы имеем информацию о количестве присутствующего товара в магазине в конце дня.

В результате работы цикла «по дням» по завершению месяца мы имеем информацию о реально присутствующем товаре и сравниваем ее с полученными нами данными. Результат их разности - количество украденного товара.

Цикл идущий по месяцам в конце каждого года выгружает собранную статистику в словарь shopResDict, где ключ – это год за который собрана статистика, а значение – это массив вида [яблок\_продано, яблок\_украдено, ручек\_продано, ручек\_украдено].

После обработки магазина мы сохраняем shopResDict текущего магазина в shopResesMass – массив хранящий shopResDict для всех магазинов штата.

В конце обработки штата мы суммируем все значения для всех магазинов штата и получаем структуру вида (имя\_штата, [яблок\_продано, яблок\_украдено, ручек\_продано, ручек\_украдено]) которую записываем в resAllStat. После обработки всех штатов мы выводим resAllStat

**Работу выполнили**

Коротчук Анастасия, Евенко Алеся, Ефарова Дарья -  
совместная работа над кодом и README